Online Stock Span (View)

Design an algorithm that collects daily price quotes for some stock and returns **the span** of that stock's price for the current day.

The **span** of the stock's price today is defined as the maximum number of consecutive days (starting from today and going backward) for which the stock price was less than or equal to today's price.

- For example, if the price of a stock over the next `7` days were `[100,80,60,70,60,75,85]`, then the stock spans would be `[1,1,1,2,1,4,6]`.

Implement the `StockSpanner` class:

- `StockSpanner()` Initializes the object of the class.
- `int next(int price)` Returns the **span** of the stock's price given that today's price is `price`.

Example 1:

```
Input

["StockSpanner", "next", "next", "next", "next", "next", "next", "next"]

[[], [100], [80], [60], [70], [60], [75], [85]]

Output

[null, 1, 1, 1, 2, 1, 4, 6]


Explanation

StockSpanner stockSpanner = new StockSpanner();

stockSpanner.next(100); // return 1

stockSpanner.next(80);  // return 1

stockSpanner.next(60);  // return 1

stockSpanner.next(70);  // return 2

stockSpanner.next(60);  // return 1

stockSpanner.next(75);  // return 4, because the last 4 prices (including today's
price of 75) were less than or equal to today's price.

stockSpanner.next(85);  // return 6
```

Constraints:

- `1 <= price <= 10^5`
- At most `10^4` calls will be made to `next`.